Neurocomputing 000 (2017) 1–10

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

The JaCalIVE framework for MAS in IVE: A case study in evolving modular robotics

J.A. Rincon, Emilia Garcia, V. Julian, C. Carrascosa*

Universitat Politècnica de València, Departamento de Sistemas Informáticos y Computación (DSIC), Camino de Vera s/n, Valencia, Spain

ARTICLE INFO

Article history:
Received 20 November 2015
Revised 30 May 2016
Accepted 1 August 2016
Available online xxx

Communicated by Chennai Guest Editor

Keywords:
Multi-agent systems
Virtual environments
Evolutive algorithms

ABSTRACT

This paper presents a framework specially designed for the execution and adaptation of Intelligent Virtual Environments. This framework, called JaCalIVE, facilitates the development of this kind of environments managing in an efficient and realistic way the evolution of parameters for the adaptation of the physical world. The framework includes a design method and a physical simulator which is in charge of giving the Intelligent Virtual Environment the look of the real or physical world, allowing to simulate physical phenomena such as gravity or collision detection. The paper also includes a case study which illustrates the use of the proposed framework as an evolutive algorithm which allows the automatic adaptation of modular robots.

1. Introduction

Nowadays, having software solutions at one's disposal that enforce autonomy, robustness, flexibility and adaptability of the system to develop is completely necessary. The dynamic agents organizations that auto-adjust themselves to obtain advantages from their environment seems a more than suitable technology to cope with the development of this type of systems. These organizations could appear in emergent or dynamic agent societies, such as grid domains, peer-to-peer networks or other contexts where agents dynamically group together to offer compound services as in Intelligent Virtual Environments (IVE). An IVE is a virtual environment simulating a physical (or real) world, inhabited by autonomous intelligent entities [1].

Today, this kind of applications are between the most demanded ones, not only as being the key for multi-user games such as *World Of Warcraft*[1] (with more than 7 million of users in 2013)[2] but also for immersive social networks such as *Second Life*[3] (with 36 million accounts created in its 10 years of history)[4]. It is in the development of these huge IVEs where the need of a quick and easy-to-use modeling toolkit arises.

These kinds of IVEs are addressed to a huge number of simultaneous entities, so they must be supported by highly scalable software. This software has also to be able to adapt to changes, not only of the amount of entities but also of their users needs. Technology currently used to develop this kind of products lacks of elements facilitating the adaptation and management of the system. Traditionally, this kind of applications use the client/server paradigm, but due to their features, a distributed approach such as multi-agent systems (MAS) seems to fit in the development of components that will evolve in an autonomous way and coordinated with the own environment's evolution. In the last decade, MAS technology has been successfully employed in similar large scale distributed systems such as Robocup Rescue simulation [2].

This paper presents the JaCalIVE[5] (Jason Cartago implemented Intelligent Virtual Environment) framework. It provides a method to develop this kind of IVEs along with a supporting platform to execute them. JaCalIVE is based on the MAM5 meta-model [3] which describes a method to design IVEs.

MAM5 is based in the A&A (Agent & Artifact) meta-model [4] that describes environments for MAS as populated not only by agents, but also for other entities that are called *artifacts*. The A&A meta-model promotes the modeling and engineering of agent societies and MAS environment as first-class entities. According to MAM5 meta-model, an IVE is composed of three important parts: artifacts, agents and physical simulation. Artifacts are the elements in which the environment is modeled. Agents are the IVE intelligent part. The physical simulation is in charge of giving the IVE

* Corresponding author.
E-mail address: carrasco@dsic.upv.es (C. Carrascosa).

1 http://www.eu.battle.net/wow.
2 http://www.statista.com/statistics/276601/number-of-world-of-warcraft-subscribers-by-quarter/.
3 http://www.secondlife.com.
4 http://massively.joystiq.com/2013/06/20/second-life-readies-for-10th-anniversary-celebrates-a-million-a/.
5 http://www.jacalive.gti-ia.dsic.upv.es/.

http://dx.doi.org/10.1016/j.neucom.2016.08.160
0925-2312/

the look of the real or physical world, allowing to simulate physical phenomenal such as gravity or collision detection.

In order to evaluate the proposed framework we have chosen a case study consisting in the implementation of an evolutive algorithm which allows the automatic creation of modular robots optimized for a specific task. Specifically, the implemented system simulates a genetic algorithm where robots can interact among them in order to change its shape by joining other modules or environment objects. In that sense, one simple modular robot can change its shape and create a complex robot depending on the movement required. The aim of each robot is to minimize the distance between its real movement and the movement defined by the fitness function. During the simulation robots will evolve or will be destroyed following the rules of an evolutive algorithm. The modular robots obtained at the end of the simulation will be the better adapted and the most appropriated to do the required movement.

The rest of the paper is organized as follows: Section 2 summarizes the most important related work. Section 3 describes the JaCalIVE framework. Section 4 presents the proposed case study based on the automatic evolution of modular robotics developed using JaCalIVE. Finally, Section 5 summarizes the main conclusions of this work.

2. Related work

This section summarizes the most relevant techniques and technology that the JaCalIVE framework integrates in order to design and simulate IVEs. These techniques allow JaCalIVE to develop IVEs that are realistic, complex, adaptable, and with autonomous and rational entities. First, some concepts about IVEs are presented, to continue commenting about Multi-Agent Systems concepts, as platforms and methodologies relevant to the present work. Finally, this section presents the MAM5 meta-model, as it is the starting point for the present work to model IVEs in MAS terms.

2.1. IVE

Currently, there is an increasing interest in the application of IVEs in a wide variety of domains. IVEs have been used to create advanced simulated environments [5–7] in so different domains as education [8], entertainment [9–12], e-commerce [13], health [14,15] and use to VR-based simulations [16].

One of the key features of any IVE is to offer a high level of user immersion. In order to achieve that, it is necessary that the IVE has the ability of simulating physical conditions of the real world such as gravity, friction and collisions. Besides, in order to increase the graphical realism, the physical simulators should include dynamic and static objects that inhabitat the IVE in a three-dimensional environment. A comparative analysis of some of the most important developed physical simulation tools can be found in [17]. The conclusion of this analysis shows that, of the open source engines, the *JBullet*[6] engine provided the best results overall, outperforming even some of the commercial engines. In this kind of simulators in which it is pretended to create a robotic simulation, it is important to give to the developer a physical engine. This engine allows to create constraints in the environment. These constraints pretend to create realistic simulations. It is very common to find simulators in which the physical engine is coupled to the stage of visualization. Some examples of this kind of simulators are Gazebo [18], V-REP[7], and Molecubes [19]. Gazebo uses ODE physical engine for physics simulation. Gazebo (similarly to aforementioned Stage simulator) can be controlled using the Player API and from Robot Operating System (ROS). V-REP is a general-purpose simulator that is used to create general robotics simulators. Whilst Molecubes [19] simulator is a more specific one, addressed to modular robotics (it was developed for Molecubes modular robots). It is based on *NVidia PhysX* and the visualization is implemented using the OGRE library. These simulators do not consider the use of multi agent systems. All these simulators employ either ODE or Bullet physics engine and this physical engine is associated to the 3D rendering process. As an example of IVEs populated by agents (MAS) DIVAs 4.0 [20,21] framework has to be underlined, as it allows to simulate an environment populated by agents, decoupled from them (although this environment / simulation is not based on agent-technology as in the proposal here presented).

2.2. Multi-agent systems

Until now, we have highlighted the importance of giving *realism* to IVEs, which would enable the user to have the desired level of immersion. This realism is provided by the physical simulation and 3D visualization, but this is only one part of a virtual environment. To be an IVE, a virtual environment needs to give entities with the intelligence to enhance the user's immersion.

MAS is one of the most employed artificial intelligence technique for modeling IVEs. This is mainly due to the characteristics that agents have, such as autonomy, proactivity, reactivity and sociability. But this does not mean that no other AI techniques can be used within MAS for IVE development. An agent can include as a decision-making mechanism other algorithms that improve the deliberative process such as reinforcement learning [22,23], genetic algorithms [24], markov models [25], classification [26,27], neuronal networks [28] or use any method of hybrid artificial intelligence systems [29] etc.

However, when modeling an environment it is necessary to take into account that not all the entities are agents. The A&A meta-model [30,31] describes a methodology for modeling environments using artifacts. Artifacts represent the first level of abstraction when modeling environments. This is mainly due to the clear differentiation of the entities which are in systems of this kind. This differentiation can determine which items are objects (Artifacts) and which are intelligent entities (Agents).

The BDI model (Belief - Desire - Intention) [32–34] is the most well-known and used agent model when designing intelligent agents. This model is based on logic and psychology, which creates symbolic representations of beliefs, desires and intentions of the agents. The beliefs are the information the agent has about the environment. This information can be updated at each time step or not. This obsolescence of the used information forces the agent to perform deliberative processes. Desires are the possible actions that the agent could make. This does not mean that every desire of an agent has to be performed. Finally, intentions represent the actions that the agent has decided to perform. These actions may be goals that have been delegated to the agent or may be the result of previous deliberation processes.

Different approaches have been devised in order to develop MAS. One of the first tools used for implementing agents is the JADE platform. JADE has been used for the development of *JGOMAS (Game Oriented Multi -Agent System based on Jade)* [35–37]. JADE does not directly provide a BDI model but there is an extension called *JADEX* allowing developers to design BDI-oriented MAs incorporating the representation of beliefs, desires and intentions. JADEX has been used for modeling environments like the presented in [38]. *Jason* is another development tool used for MAS programming which also integrates the BDI model.

In our proposal we employ Jason as the programming toolkit for our BDI agents [39]. The main reason to employ JASON is its full integration with CArtAgO (Common ARTifact

[6] jbullet.advel.cz/.
[7] www.coppeliarobotics.com/.

Please cite this article as: J.A. Rincon et al., The JaCalLive framework for MAS in IVE: A case study in evolving modular robotics, Neurocomputing (2017), http://dx.doi.org/10.1016/j.neucom.2016.08.160